

What is claimed is:

- 1     1.     A machine-implemented method, comprising:  
2             executing a first instance of a virtual console driver that is implemented by an  
3                 operating system kernel instance;  
4             establishing a first device node within a first virtual operating system environment  
5                 (VOSE) of a plurality of VOSEs controlled by the operating system kernel  
6                 instance;  
7             establishing an association between the first device node and the first instance of the  
8                 virtual console driver;  
9             in response to a first process' request to write to the first device node, determining  
10                 with which instance of the virtual console driver the first device node is  
11                 associated, wherein the first process executes in the first VOSE; and  
12             in response to determining that the first device node is associated with the first  
13                 instance of the virtual console driver, sending, to the first instance of the  
14                 virtual console driver, data received from the first process.
- 1     2.     The method of claim 1, wherein, except for processes executing in the first VOSE,  
2             the first device node is not accessible by any processes executing in any VOSE of the  
3             plurality of VOSES.
- 1     3.     The method of claim 1, wherein the first device node is exposed as “/dev/console” to  
2             processes executing in the first VOSE.

4. The method of claim 1, further comprising:  
 establishing a second device node within a global operating system environment  
 (OSE) that comprises the plurality of VOSEs;  
 establishing an association between the second device node and the first instance of  
 the virtual console driver;  
 in response to a second process' request to read from the second device node,  
 determining with which instance of the virtual console driver the second  
 device node is associated, wherein the second process executes in the global  
 OSE; and  
 in response to determining that the second device node is associated with the first  
 instance of the virtual console driver, sending, to the second process, the data  
 that was received from the first process and sent to the first instance of the  
 virtual console driver.

5. The method of claim 5, wherein the second device node is not accessible by any  
 processes executing in any VOSE of the plurality of VOSES.

6. The method of claim 5, further comprising:  
 receiving, from the second process, a command to execute an instance of the virtual  
 console driver;  
 wherein the step of executing the first instance of the virtual console driver is  
 performed in response to receiving the command from the second process.

7. The method of claim 5, further comprising:  
 receiving, from the second process, a command to establish the first device node  
 within the first VOSE;

wherein the step of establishing the first device node within the first VOSE is performed in response to receiving the command from the second process.

8. The method of claim 1, further comprising:

executing a second instance of the virtual console driver, wherein the second instance of the virtual console driver is separate from the first instance of the virtual console driver;

establishing a second device node within a second virtual operating system environment (VOSE) of the plurality of VOSEs, wherein the second VOSE is separate from the first VOSE;

establishing an association between the second device node and the second instance of the virtual console driver;

in response to a second process' request to write to the second device node, determining with which instance of the virtual console driver the second device node is associated, wherein the second process executes in the second VOSE; and

in response to determining that the second device node is associated with the second instance of the virtual console driver, sending, to the second instance of the virtual console driver, data received from the second process.

9. The method of claim 8, wherein:

except for processes executing in the first VOSE, the first device node is not accessible by any processes executing in any VOSE of the plurality of VOSES; and

5           except for processes executing in the second VOSE, the second device node is not  
6           accessible by any processes executing in any VOSE of the plurality of  
7           VOSES.

1    10.    The method of claim 8, wherein:  
2           the first device node is exposed as “/dev/console” to processes executing in the first  
3           VOSE; and  
4           the second device node is exposed as “/dev/console” to processes executing in the  
5           second VOSE.

1    11.    A machine-readable medium, comprising:  
2           instructions for causing one or more processors to execute a first instance of a virtual  
3           console driver that is implemented by an operating system kernel instance;  
4           instructions for causing one or more processors to establish a first device node within  
5           a first virtual operating system environment (VOSE) of a plurality of VOSEs  
6           controlled by the operating system kernel instance;  
7           instructions for causing one or more processors to establish an association between  
8           the first device node and the first instance of the virtual console driver;  
9           instructions for causing one or more processors to determine, in response to a first  
10          process’ request to write to the first device node, with which instance of the  
11          virtual console driver the first device node is associated, wherein the first  
12          process executes in the first VOSE; and  
13          instructions for causing one or more processors to send data received from the first  
14          process to the first instance of the virtual console driver in response to

15 determining that the first device node is associated with the first instance of  
16 the virtual console driver.

1 12. The machine-readable medium of claim 11, wherein, except for processes executing  
2 in the first VOSE, the first device node is not accessible by any processes executing  
3 in any VOSE of the plurality of VOSES.

1 13. The machine-readable medium of claim 11, wherein the first device node is exposed  
2 as “/dev/console” to processes executing in the first VOSE.

1 14. The machine-readable medium of claim 11, further comprising:  
2 instructions for causing one or more processors to establish a second device node  
3 within a global operating system environment (OSE) that comprises the  
4 plurality of VOSEs;  
5 instructions for causing one or more processors to establish an association between  
6 the second device node and the first instance of the virtual console driver;  
7 instructions for causing one or more processors to determine, in response to a second  
8 process’ request to read from the second device node, with which instance of  
9 the virtual console driver the second device node is associated, wherein the  
10 second process executes in the global OSE; and  
11 instructions for causing one or more processors to send, to the second process, in  
12 response to determining that the second device node is associated with the  
13 first instance of the virtual console driver, the data that was received from the  
14 first process and sent to the first instance of the virtual console driver.

1 15. The machine-readable medium of claim 14, wherein the second device node is not  
2 accessible by any processes executing in any VOSE of the plurality of VOSES.

1    16.    The machine-readable medium of claim 14, further comprising:  
2           instructions for causing one or more processors to receive, from the second process, a  
3           command to execute an instance of the virtual console driver;  
4           wherein the instructions for causing one or more processors to execute the first  
5           instance of the virtual console driver comprise instructions for causing one or  
6           more processors to execute the first instance of the virtual console driver in  
7           response to receiving the command from the second process.

1    17.    The machine-readable medium of claim 14, further comprising:  
2           instructions for causing one or more processors to receive, from the second process, a  
3           command to establish the first device node within the first VOSE;  
4           wherein the instructions for causing one or more processors to establish the first  
5           device node within the first VOSE comprise instructions for causing one or  
6           more processors to establish the first device node within the first VOSE in  
7           response to receiving the command from the second process.

1    18.    The machine-readable medium of claim 11, further comprising:  
2           instructions for causing one or more processors to execute a second instance of the  
3           virtual console driver, wherein the second instance of the virtual console  
4           driver is separate from the first instance of the virtual console driver;  
5           instructions for causing one or more processors to establish a second device node  
6           within a second virtual operating system environment (VOSE) of the plurality  
7           of VOSEs, wherein the second VOSE is separate from the first VOSE;  
8           instructions for causing one or more processors to establish an association between  
9           the second device node and the second instance of the virtual console driver;

10 instructions for causing one or more processors to determine, in response to a second  
 11 process' request to write to the second device node, with which instance of the  
 12 virtual console driver the second device node is associated, wherein the  
 13 second process executes in the second VOSE; and  
 14 instructions for causing one or more processors to send data received from the second  
 15 process to the second instance of the virtual console driver in response to  
 16 determining that the second device node is associated with the second instance  
 17 of the virtual console driver.

1 19. The machine-readable medium of claim 18, wherein:  
 2 except for processes executing in the first VOSE, the first device node is not  
 3 accessible by any processes executing in any VOSE of the plurality of  
 4 VOSES; and  
 5 except for processes executing in the second VOSE, the second device node is not  
 6 accessible by any processes executing in any VOSE of the plurality of  
 7 VOSES.

1 20. The machine-readable medium of claim 18, wherein:  
 2 the first device node is exposed as “/dev/console” to processes executing in the first  
 3 VOSE; and  
 4 the second device node is exposed as “/dev/console” to processes executing in the  
 5 second VOSE.

1 21. An apparatus, comprising:  
 2 a mechanism for executing a first instance of a virtual console driver that is  
 3 implemented by an operating system kernel instance;

4 a mechanism for establishing a first device node within a first virtual operating  
5 system environment (VOSE) of a plurality of VOSEs controlled by the  
6 operating system kernel instance;  
7 a mechanism for establishing an association between the first device node and the  
8 first instance of the virtual console driver;  
9 a mechanism for determining, in response to a first process' request to write to the  
10 first device node, with which instance of the virtual console driver the first  
11 device node is associated, wherein the first process executes in the first  
12 VOSE; and  
13 a mechanism for sending data received from the first process to the first instance of  
14 the virtual console driver in response to determining that the first device node  
15 is associated with the first instance of the virtual console driver.

1 22. The apparatus of claim 21, wherein, except for processes executing in the first VOSE,  
2 the first device node is not accessible by any processes executing in any VOSE of the  
3 plurality of VOSES.

1 23. The apparatus of claim 21, wherein the first device node is exposed as “/dev/console”  
2 to processes executing in the first VOSE.

1 24. The apparatus of claim 21, further comprising:  
2 a mechanism for establishing a second device node within a global operating system  
3 environment (OSE) that comprises the plurality of VOSEs;  
4 a mechanism for establishing an association between the second device node and the  
5 first instance of the virtual console driver;



6           a mechanism for determining, in response to a second process' request to read from  
 7           the second device node, with which instance of the virtual console driver the  
 8           second device node is associated, wherein the second process executes in the  
 9           global OSE; and  
 10          a mechanism for sending the data that was received from the first process and sent to  
 11          the first instance of the virtual console driver to the second process in  
 12          response to determining that the second device node is associated with the  
 13          first instance of the virtual console driver.

1   25.    The apparatus of claim 24, wherein the second device node is not accessible by any  
 2           processes executing in any VOSE of the plurality of VOSES.

1   26.    The apparatus of claim 24, further comprising:  
 2           a mechanism for receiving, from the second process, a command to execute an  
 3           instance of the virtual console driver;  
 4           wherein the mechanism for executing the first instance of the virtual console driver  
 5           comprises a mechanism for executing the first instance of the virtual console  
 6           driver in response to receiving the command from the second process.

1   27.    The apparatus of claim 24, further comprising:  
 2           a mechanism for receiving, from the second process, a command to establish the first  
 3           device node within the first VOSE;  
 4           wherein the mechanism for establishing the first device node within the first VOSE  
 5           comprises a mechanism for establishing the first device node within the first  
 6           VOSE in response to receiving the command from the second process.

1   28.    The apparatus of claim 21, further comprising:

a mechanism for executing a second instance of the virtual console driver, wherein  
the second instance of the virtual console driver is separate from the first  
instance of the virtual console driver;  
a mechanism for establishing a second device node within a second virtual operating  
system environment (VOSE) of the plurality of VOSEs, wherein the second  
VOSE is separate from the first VOSE;  
a mechanism for establishing an association between the second device node and the  
second instance of the virtual console driver;  
a mechanism for determining, in response to a second process' request to write to the  
second device node, with which instance of the virtual console driver the  
second device node is associated, wherein the second process executes in the  
second VOSE; and  
a mechanism for sending data received from the second process to the second  
instance of the virtual console driver in response to determining that the  
second device node is associated with the second instance of the virtual  
console driver.

29. The apparatus of claim 28, wherein:

except for processes executing in the first VOSE, the first device node is not  
accessible by any processes executing in any VOSE of the plurality of  
VOSES; and  
except for processes executing in the second VOSE, the second device node is not  
accessible by any processes executing in any VOSE of the plurality of  
VOSES.

- 1    30.    The apparatus of claim 28, wherein:  
2            the first device node is exposed as “/dev/console” to processes executing in the first  
3            VOSE; and  
4            the second device node is exposed as “/dev/console” to processes executing in the  
5            second VOSE.